AMENDMENTS TO THE CLAIMS

Please amend the claims of the present application as set forth in the listing of claims below. This listing of claims will replace all prior versions, and listings, of claims in the application.

Claims 1-52 were originally filed.

Claims 45 – 52 have been withdrawn.

Claims 7 - 14, 21, and 27 have been cancelled.

Claims 1, 2, 15, 36, and 38 have been amended.

Accordingly, claims 1 – 6, 15 – 20, 22 – 26, and 28 - 44 are pending.

LISTING OF CLAIMS

1.   **(Currently Amended)**   A method comprising:

receiving a request to write data to a logical sector address of a flash memory medium;

selecting a physical sector address from a list of free physical sector addresses;

assigning [[a]] the selected free physical sector address to the logical sector address forming a corresponding relationship between the addresses;

storing the corresponding relationship between the addresses in a data structure; and

writing the data into a physical sector of the flash memory medium at a location indicated by the selected free physical sector address.

2.   **(Currently Amended)**   The method as recited in Claim 1, wherein the data structure is ~~contained in at least one memory device other than~~ <u>stored on</u> the flash memory medium.

3.   **(Original)**   The method as recited in Claim 1, wherein the data structure is contained in a random access memory device.

4.   **(Original)**   The method as recited in Claim 1, wherein the request is received from a file system.

5.   **(Original)**   The method as recited in Claim 1, further comprising storing the logical sector address in the physical sector of the flash memory medium along with the data.

6.   **(Original)**   The method as recited in Claim 1, further comprising:

storing the logical sector address in the physical sector of the flash memory medium along with the data;

if the data structure is erased, then scanning the flash memory medium to locate the logical sector address stored with the data;

assigning the physical sector address containing the data to the logical

sector address forming a reestablished corresponding relationship between the

addresses; and

storing the reestablished corresponding relationship between the addresses

in a new data structure.

7 - 14. **(Canceled)**

15.   **(Currently Amended)**   A method comprising A computer-readable

medium having computer-executable instructions for performing steps

comprising:

receiving a request to write data to a logical sector address of a flash

memory medium;

selecting a physical sector of the flash memory medium to store the data

based on the ability of the physical sector to store the data without first being

erased;

assigning a physical sector address of the selected physical sector to the

logical sector address forming a corresponding relationship between the addresses;

storing the corresponding relationship between the addresses in a data

structure;

writing the data into a the physical sector of the flash memory medium at a

location indicated by the physical sector address; and

writing the logical sector address in the physical sector of the flash memory medium along with the data.

16. **(Original)**   The method as recited in Claim 15, further comprising:

if the data structure is erased, then scanning the flash memory medium to locate the logical sector address stored with the data;

assigning the physical sector address containing the data to the logical sector address forming a reestablished corresponding relationship between the addresses; and

storing the reestablished corresponding relationship between the addresses in a new data structure.

17. **(Original)**   The method as recited in Claim 15, wherein writing the logical sector address in the physical sector of the flash memory medium includes writing the logical sector address into a spare portion of the physical sector.

18. **(Original)**   The method as recited in Claim 15, wherein the data structure is contained in at least one memory device other than the flash memory medium.

19. **(Original)**   The method as recited in Claim 15, wherein the data structure is contained in a random access memory device.

20. **(Original)** The method as recited in Claim 15, wherein the request is received from a file system.

21. **(Canceled)**

22. **(Original)** A method comprising:

(a) receiving a request to write data to a logical sector address of a flash memory medium;

(b) assigning a physical sector address to the logical sector address forming a corresponding relationship between the addresses;

(c) storing the corresponding relationship between the addresses in a data structure;

(d) writing the data into a physical sector of the flash memory medium at a location indicated by the physical sector address;

(e) receiving a request to rewrite updated data to the logical sector address;

(f) assigning a new physical sector address to the logical sector address forming a corresponding relationship between the new physical sector address and the logical sector address;

(g) storing the corresponding relationship between the addresses from the aforementioned paragraph (f) in the data structure;

(h)   writing the updated data into a physical sector of the flash memory medium at a location indicated by the new physical sector address; and

(i)   marking the physical sector address from the aforementioned paragraph (b) as dirty.

23.   **(Original)**   The method as recited in Claim 22, wherein the data structure is contained in at least one memory device other than the flash memory medium.

24.   **(Original)**   The method as recited in Claim 22, wherein the data structure is contained in a random access memory device.

25.   **(Original)**   The method as recited in Claim 22, wherein the requests are received from a file system.

26.   **(Original)**   The method as recited in Claim 22, wherein the data structure is maintained by a flash abstraction logic of a flash memory driver.

27.   **(Canceled)**

28.   **(Original)**   A system, comprising:

flash medium logic, configured to store data in a physical sector of a flash memory medium;

a table, configured to map logical sector addresses received from a file system to physical sector addresses on the flash memory medium; and

flash abstraction logic, configured to ascertain a next free physical sector on a flash memory medium and assign an address associated with the free physical sector to a logical sector address associated with a write request received from the file system.

29.   **(Original)**   The system as recited in Claim 28, wherein the flash abstraction logic is further configured to update the map of the logical sector addresses to the physical sector addresses, after assigning the address associated with the free physical sector to the logical sector address associated with the write request.

30.   **(Original)**   The system as recited in Claim 28, wherein the flash medium logic marks a physical sector address as dirty after contents associated with the physical sector address are rewritten to a new physical sector address.

31.   **(Original)**   The system as recited in Claim 28, wherein the flash medium logic is further configured to store the logical sector address within a portion of the physical sector associated with the write request.

32.   **(Original)**   The system as recited in Claim 28, wherein the flash medium logic is further configured to:

store the logical sector address within a portion of the physical sector associated with the write request; and

scan the flash memory medium for the portion of the physical sector storing the logical sector address after initialization of the system.

33.   **(Original)**   The system as recited in Claim 28, wherein the table is contained in a memory device other than the flash memory medium.

34.   **(Original)**   The system as recited in Claim 28, wherein the table is contained in a random access memory device.

35.   **(Original)**   The system as recited in Claim 28, wherein the system is a flash driver system.

36.   **(Currently Amended)**   A computer-readable medium for a flash driver, comprising computer-executable instructions that, when executed, direct the flash driver to:

receive a request to write data to a logical sector address of a flash memory medium;

selecting a physical sector address from a list of free physical sector addresses;

assign [[a]] the selected physical sector address to the logical sector address forming a corresponding relationship between the addresses;

store the corresponding relationship between the addresses in a table; and

write the data into a physical sector of the flash memory medium at a location indicated by the physical sector address.

37.   **(Original)**   A computer-readable medium for a flash driver, comprising computer-executable instructions that, when executed, direct the flash driver to:

receive a request to write data to a logical sector address of a flash memory medium;

assign a physical sector address to the logical sector address forming a corresponding relationship between the addresses;

store the corresponding relationship between the addresses in a table;

write the data into a physical sector of the flash memory medium at a location indicated by the physical sector address;

write the logical sector address in the physical sector of the flash memory medium along with the data;

if the table is erased, then scan the flash memory medium to locate the logical sector address stored with the data;

assign the physical sector address containing the data to the logical sector address forming a reestablished corresponding relationship between the addresses; and

store the reestablished corresponding relationship between the addresses in a new table.

38.   **(Currently Amended)**   A system for tracking sectors in a flash memory medium, comprising:

means for receiving a request to retrieve data stored in the flash memory medium from a location indicated by a logical sector address;

means for locating a physical sector address corresponding to the specific logical sector address ~~from~~ <u>in</u> a table; and

means for reading the data stored in the flash memory medium from the physical sector address ~~retrieved~~ <u>located</u> from the table.

39.   **(Original)**   The system as recited in Claim 38, further comprising:

means for storing the logical sector address with the data in the flash memory medium at a location indicated by the physical sector address;

means for reestablishing a portion of the table in the event of a power interruption, by scanning the physical sector address for the associated logical sector address; and

means for storing the logical sector address in the table at a location corresponding to the physical sector address.

40.   **(Original)**   The system as recited in Claim 38, wherein the request is received from a file system.

42.   **(Original)**   The system as recited in Claim 38, wherein the table is maintained by a flash abstraction logic of a flash driver.

43.   **(Original)**   The system as recited in Claim 38, wherein the means for reading of the data is performed by flash media logic of a flash driver.

44.   **(Original)**   The system as recited in Claim 38, wherein the memory device is a type of random access memory.

45.   **(Withdrawn)**   A flash driver system, comprising:

a free sector manager, configured to determine a next free physical sector address on the media and assign the address to a logical sector address of a write request received from a file system;

a table, configured to store a map showing the assignment of the physical sector address to the logical sector address; and

a flash medium logic, configured to write the data to the next free physical sector indicated by the free sector manager and store the logical sector address directly with the data on the flash memory medium.

46.   **(Withdrawn)**   The flash driver as recited in Claim 45, further comprising a flash abstraction logic, configured to scan the sectors of the flash memory medium for the logical sector address and note the physical sector address from which the logical sector address is stored and reestablish the map in the table in the event the table is erased.

47.   **(Withdrawn)**   The flash driver as recited in Claim 45, wherein the logical sector address is stored in a spare portion of the flash memory medium.

48.   **(Withdrawn)**   A flash driver system, comprising:

a free sector manager, configured to determine a next free physical sector address available on the flash memory medium;

a flash abstraction logic, configured to query the free sector manager for the next free physical sector address and link the physical sector address to a logical sector address received from a file system; and

a table, configured to store the physical sector address to logical sector address linking performed by the flash abstraction logic.

49.	**(Withdrawn)**	The flash driver system as recited in Claim 48, wherein the flash abstraction logic is further configured to mark an existing physical sector as dirty, if a logical sector address received from the file system was previously linked to the logical sector address in the table.

50.	**(Withdrawn)**	The flash driver system as recited in Claim 48, wherein the flash abstraction logic is further configured to mark an existing physical sector as dirty, if a logical sector address received from the file system was previously linked to the logical sector address in the table, but only after data associated with the logical sector address is successfully written to a new physical sector address on the flash memory media.

51.	**(Withdrawn)**	The flash driver system as recited in Claim 48, wherein a portion of a physical sector is a status bit configured to indicate when a write operation is in progress.

52.	**(Withdrawn)**	The flash driver system as recited in Claim 48, wherein a portion of physical sector is a status bit configured to indicate when a write operation has been completed successfully.